Securing your Mikrotik Network

Andrew Thrift
andrew@networklabs.co.nz
Who am I?

Andrew Thrift

Mikrotik Certified Consultant

Based in New Zealand

Using Mikrotik RouterOS since around 2002

Working in network security since 1999

Blog with Andrew Cox @ www.mikrotik-routeros.com
Agenda

- Attack Types
- Detecting attacks
- Securing your routers
- Protecting your network
- Question Time
Types of network based attacks

• Attacks on your routers
  ▪ Unauthorised logins
  ▪ Brute force attacks
  ▪ Denial Of Service

• Customer misuse
  ▪ Customers bypassing PPPoE server
  ▪ Rogue DHCP Servers

• Attacks on your networks (customers)
  ▪ Brute force attacks
  ▪ Denial Of Service
Detecting Attacks

Use Intrusion Detection System (IDS/IPS) software
- Snort / Suricata
- Place behind your “border” protection
- Configure alerting

Use Security Information Events Management (SIEM) software
- Sagan
What is an IDS/IPS

Intrusion Detection System

- Inspects network traffic for “known threats”
- Identifies network threats using:
  - Signatures
  - Behavioural Analysis
  - Heuristics
- Ranks risk severity “Low, Medium, High”
- Common IDS are Snort, Suricata, Bro-IDS

Intrusion Prevention System

- Same as an IDS, but is placed “Inline” and can take actions (drop/mark) based on risk.
Detecting Attacks – Intrusion Detection System

The Internet

Border Router

Layer2 Mirror Port/Tap

Your Networks

Intrusion Detection System Snort/Suricata
IDS/IPS – What is Suricata

Heard of Snort?
IDS/IPS - Suricata

Suricata is like Snort, but is better:

- Multi-threaded to scale better on Multi-Core, Multi-Processor systems
- More sane configuration
- Can use existing Snort rule bases
- Fully supported by Emergingthreats.net standard and pro rule bases
- Has been demonstrated doing IPS at wire speed 25 Gigabit on Tilera processors
What is a SIEM?

Security Information & Events Management

- Inspects log entries and correlates these to “known threats”

- Identifies network threats using:
  - Signatures
  - Behavioural Analysis
  - Heuristics

- Ranks risk severity “Low, Medium, High”

- Common SIEM are Sagan, OSSIM

- Generally require custom rules for RouterOS
Sagan is a log analyser:

- Analyses log traffic sent to it via Syslog
- Multi-threaded – Scales well on multi-core/multi-processor systems
- Has flexible “rules” that can correlate multiple different events into a security event.
- Outputs in Snort format allowing for easy integration
Snorby

Provides a nice Web interface to analyse Suricata + Sagan results
Intrusion Detection for the lazy

Ubuntu + Suricata + Snorby = SmoothSec

• Pre-Built “Appliance”
• Works out of the box
• Available from http://bailey.st/blog/smooth-sec/
• Can apt-get install sagan for SIEM functionality 😊
What else can you do with an IPS?

Accurately detect difficult protocols e.g.

- BitTorrent (including DHT/Trackerless torrents)
- Skype (Signalling and media)
- Youtube (Native and embedded)
- VoIP (Signalling and media)

This can be done on standard and non-standard ports.

On match the IPS can change the DSCP tag. Your Mikrotik router can then identify the traffic in mangle using the DSCP tag, and you can then queue this traffic appropriately.
Protecting your routers

Mikrotik Routers have no security configured by default. There are NO firewall policies, all services are accessible from everywhere. You need to protect yourself or it is only a matter of time before your routers are compromised.

How?

• Disable unused services (WinBox, Telnet, SSH, WebMin)

• Implement “input” IP filters to:
  ▪ Restrict access to router management
  ▪ Minimise the impact of Denial of Service type attack

• Only allow management access within a dedicated Management VRF (RouterOS 6.x + New Routing package)
Disabling the services you do not use is easy, and once disable these can not be attacked.

To disable IP services, simply go to: **IP → Services** in Winbox and disable the services you do not need.
Protecting your routers – IP Filtering

Create “input” policies, accepting the protocols you need. E.g. Winbox, SSH, BGP, OSPF, MPLS LDP, PPTP, DNS. Be specific in your policies, only allow these protocols to enter via a specific interface, or use “Address Lists” to limit these to originate from a group of your subnets.

---

Firewall

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>jump</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>190.2 MiB</td>
<td>5 478 999</td>
<td>Accept ICMP requ...</td>
</tr>
<tr>
<td>1</td>
<td>drop</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3300.4 KiB</td>
<td>68 409</td>
<td>Drop Invalid</td>
</tr>
<tr>
<td>2</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>64.4 GiB</td>
<td>301 665</td>
<td>Accept Established</td>
</tr>
<tr>
<td>3</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>55.8 GiB</td>
<td>602 908</td>
<td>Accept Related</td>
</tr>
<tr>
<td>4</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>270.5 MiB</td>
<td>4 574 652</td>
<td>Accept LDP from ...</td>
</tr>
<tr>
<td>5</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2940 B</td>
<td>49</td>
<td>Accept LDP from ...</td>
</tr>
<tr>
<td>6</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>170.4 MiB</td>
<td>2 475 852</td>
<td>Accept OSPF from ...</td>
</tr>
<tr>
<td>7</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>105.0 MiB</td>
<td>913 644</td>
<td>Accept Neighbor ...</td>
</tr>
<tr>
<td>8</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>107.7 MiB</td>
<td>1 485 526</td>
<td>Accept NTP traffic...</td>
</tr>
<tr>
<td>9</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0 B</td>
<td>0</td>
<td>Accept BGP from ...</td>
</tr>
<tr>
<td>10</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>975.6 KiB</td>
<td>666</td>
<td>Accept BTest from...</td>
</tr>
<tr>
<td>11</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1966.3 KiB</td>
<td>28 084</td>
<td>Accept DNS quer...</td>
</tr>
<tr>
<td>12</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>196.7 MiB</td>
<td>1 202 033</td>
<td>Accept DNS queri...</td>
</tr>
<tr>
<td>13</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>156 B</td>
<td>0</td>
<td>3 Accept WinBox tra...</td>
</tr>
<tr>
<td>14</td>
<td>accept</td>
<td>input</td>
<td>131.203.25...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>accept</td>
<td>input</td>
<td>10.0.0.0/8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>accept</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>drop</td>
<td>input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2078.5 MiB</td>
<td>14 450 068</td>
<td>Drop EVERYTHING...</td>
</tr>
</tbody>
</table>
## Protecting your routers – Common Services

<table>
<thead>
<tr>
<th>Service</th>
<th>Protocol/Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Winbox</td>
<td>TCP 8291</td>
</tr>
<tr>
<td>SSH</td>
<td>TCP 22</td>
</tr>
<tr>
<td>Telnet</td>
<td>TCP 23</td>
</tr>
<tr>
<td>Webmin</td>
<td>TCP 80 and TCP 443</td>
</tr>
<tr>
<td>OSPF</td>
<td>OSPF (Protocol 89)</td>
</tr>
<tr>
<td>BGP</td>
<td>TCP 179</td>
</tr>
<tr>
<td>MPLS LDP</td>
<td>TCP 646 and UDP 646</td>
</tr>
<tr>
<td>Neighbor Discovery</td>
<td>UDP 5678</td>
</tr>
<tr>
<td>Btest</td>
<td>UDP 2250-2270</td>
</tr>
</tbody>
</table>
Protecting your network

As well as protecting your routers from attack, you may want to protect your clients from attacks such as:

- Distributed Denial of Service (DDoS)
- Brute Force
- ICMP flooding

And your network from:

- Unauthorised transit
- Customer misuse
Protecting your network – ICMP (Ping) Flooding

This example shows the limiting of ICMP traffic. This works by allowing the various types of ICMP traffic at a rate of up to 5 packets a second. If ICMP traffic exceeds this, then it will be dropped.

**NOTE:** This policy will need tuning if you are using it in your “forward” chain.

This policy can be used as-is, for protecting your router in the “input” chain.

```Shell
/ip firewall filter
add chain=forward protocol=icmp action=jump jump-target=ICMP comment="jump to chain ICMP" disabled=no
add chain=ICMP protocol=icmp icmp-options=0:0-255 limit=5,5 action=accept comment="0:0 and limit for 5 pac/s" disabled=no
add chain=ICMP protocol=icmp icmp-options=3:3 limit=5,5 action=accept comment="3:3 and limit for 5 pac/s" disabled=no
add chain=ICMP protocol=icmp icmp-options=3:4 limit=5,5 action=accept comment="3:4 and limit for 5 pac/s" disabled=no
add chain=ICMP protocol=icmp icmp-options=8:0-255 limit=5,5 action=accept comment="8:0 and limit for 5 pac/s" disabled=no
add chain=ICMP protocol=icmp icmp-options=11:0-255 limit=5,5 action=accept comment="11:0 and limit for 5 pac/s" disabled=no
add chain=ICMP protocol=icmp action=drop comment="Drop everything else" disabled=no
```

- **Send ICMP traffic to the “ICMP” chain**
- **Allow ICMP up to 5 packets a second**
- **Drop any remaining ICMP traffic**
Protecting your network – SSH brute force

This example shows protecting your customers from SSH brute force attacks.

It works by adding the Source IP of the party originating the SSH session to an address list, if this Source party starts another SSH session within a 1 minute timeframe it escalates it up to the next level of address list. If the source party continues to create new SSH sessions, they will be escalated to the “ssh_blacklist” and will not be able to create SSH sessions for 10 days.

```
/ip firewall filter
add chain=forward protocol=tcp dst-port=22 src-address-list=ssh_blacklist action=drop \ comment="drop ssh brute force attempts" disabled=no

add chain=forward protocol=tcp dst-port=22 connection-state=new \ src-address-list=ssh_stage1 action=add-src-to-address-list address-list=ssh_stage2 \ address-list-timeout=1m comment="" disabled=no

add chain=forward protocol=tcp dst-port=22 connection-state=new \ src-address-list=ssh_stage2 action=add-src-to-address-list address-list=ssh_blacklist \ address-list-timeout=10d comment="" disabled=no

add chain=forward protocol=tcp dst-port=22 connection-state=new \ src-address-list=ssh_stage3 action=add-src-to-address-list address-list=ssh_blacklist \ address-list-timeout=10d comment="" disabled=no

add chain=forward protocol=tcp dst-port=22 connection-state=new src-address-list=ssh_stage1 \ action=add-src-to-address-list address-list=ssh_stage2 address-list-timeout=1m comment="" disabled=no

add chain=forward protocol=tcp dst-port=22 connection-state=new action=add-src-to-address-list \ address-list=ssh_stage1 address-list-timeout=1m comment="" disabled=no
```

Drop's traffic from any IP in the "ssh_blacklist"

If the IP is in "ssh_stage3" and this is a new connection, add the IP to "ssh_blacklist" with a 10 day timeout

If the IP is in "ssh_stage2" and this is a new connection, add the IP to "ssh_stage3"

If the IP is in "ssh_stage1" and this is a new connection, add the IP to "ssh_stage2"

Add the IP of SSH src to "ssh_stage1" address list if the connection is new

These same techniques can be used for numerous other protocols.
What is unauthorised transit?

Unauthorised transit is when another party uses your routers to provide transit.

This is common on Internet Exchanges. Your routers will trust the 3rd party as their prefixes will have been received from the trusted IX, the 3rd party will then route traffic via your router which will route it to one of your transit providers.

The 3rd party could now be getting internet bandwidth via your network, at your cost.
Detecting unauthorised transit

This can be done by increasing “visibility” in to your network.

• Use torch on egress port

• Use sflow + analytics software (NTOP/Scruitinizer/Solar Winds)

Look for Source addresses that are not within your IP ranges
Preventing unauthorised transit

This can be prevented by restricting L3 forwarding, and controlling your BGP advertisements.

Restrict Layer3 transit (routing) of any networks that are NOT our own

Create an address list containing your subnets

Create 4 IP filter policies

Accept traffic originating from our subnets, to our subnets

Accept traffic originating from our subnets to the Internet

Accept traffic from the Internet to our subnets

Drop all other traffic attempting to forward
Preventing unauthorised transit

Do NOT advertise prefixes that are NOT your own to upstream BGP peers

Create filter for your upstream peer(s)

- Accept your subnets
- Discard everything else

<table>
<thead>
<tr>
<th>#</th>
<th>Chain</th>
<th>Prefix</th>
<th>Prefix Length</th>
<th>Protocol</th>
<th>BGP AS Path</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>as9503-out</td>
<td>114.31.212.0/24</td>
<td></td>
<td></td>
<td></td>
<td>accept</td>
</tr>
<tr>
<td>1</td>
<td>as9503-out</td>
<td>114.31.213.0/24</td>
<td></td>
<td></td>
<td></td>
<td>accept</td>
</tr>
<tr>
<td>2</td>
<td>as9503-out</td>
<td>114.31.214.0/24</td>
<td></td>
<td></td>
<td></td>
<td>accept</td>
</tr>
<tr>
<td>3</td>
<td>as9503-out</td>
<td>114.31.215.0/24</td>
<td></td>
<td></td>
<td></td>
<td>accept</td>
</tr>
<tr>
<td>4</td>
<td>as9503-out</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>log</td>
</tr>
<tr>
<td>5</td>
<td>as9503-out</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>discard</td>
</tr>
</tbody>
</table>
BOGON Filtering

A **BOGON** is a Bogus IP address.

BOGON lists contain ranges of IP addresses that are known to have not been allocated by the Regional Internet Registries (APNIC) for use.

These are often used for malicious purposes. BOGON lists can be used on Border Routers as a first line of defence, and can reduce the effect of DOS attacks as well as incoming spam and network scans.

As Regional Internet Registry allocations are constantly changing, BOGON lists should not be static.
Using Team CYMRU BOGON BGP feed

1. Request a CYMRU BGP peering session, see www.team-cymru.org

2. Configure your Mikrotik router to peer with Team CYMRU AS65332 (use a loopback!)

3. Configure a routing filter to turn all routes received from CYMRU community 65332:888 into Black Hole routes
Success

4896 BOGON Subnets will now be blocked at the Border of our networks

\[D = Dynamic, \ A = Active, \ b = BGP, \ B = Blackhole\]
Preventing Customer Misuse – PPPoE filtering

When backhauling PPPoE to a central concentrator via VPLS/EoIP, you can prevent customers from creating their own networks by using Bridge Filters:

- Use Admin-Mac to create a static MAC on the bridge on PPPoE concentrator.
- Bridge Filter config on router closest to customer:
  - Allow pppoe-discovery to ALL destinations
  - Allow pppoe-session ONLY to PPPoE Server
  - DROP all other traffic
Preventing Customer Misuse – Rogue DHCP Servers

When operating a DHCP based network, it is common to encounter customers who run a DHCP server on their public facing interface. These are called “Rogue” DHCP servers, and can cause outages to other customers by hijacking their DHCP request and responding with settings that differ to your own DHCP server.

Luckily, this is easy to fix using Bridge Filters

- Accept Input of DHCP requests
- Accept Output of DHCP responses
- Drop forwarding of all DHCP packets
Stay Secure!

- Comments and feedback: andrew@networklabs.co.nz

- Recommended Reading:
  - wiki.mikrotik.com

- Links:
  - Suricata – http://www.openinfosecfoundation.org/
  - OSSIM - http://communities.alienvault.com/community/
  - Team CYMRU - http://www.team-cymru.org/Services/Bogons/
  - Tilera/Suricata - http://www.tilera.com/about_tilera/press-releases/tilera%E2%80%99s-tile-gx-delivers-

- Trying to identify P2P / Media? – Email me andrew@networklabs.co.nz